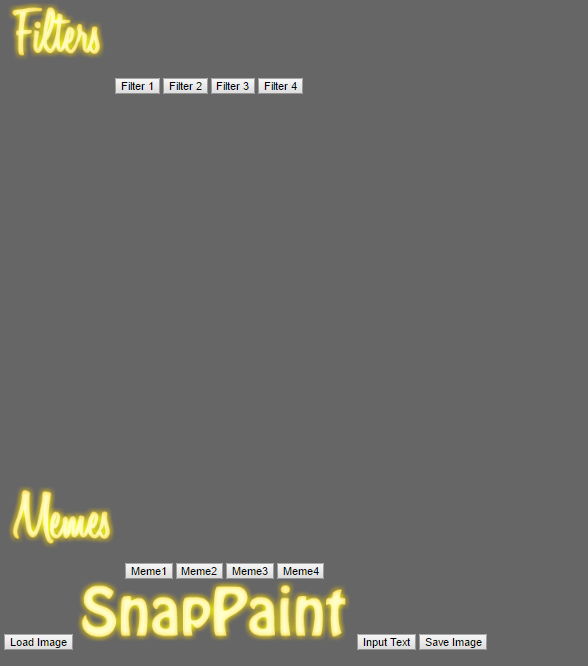
**CSPC 1045: Proposal 2**

****

**SnapPaint UI**

HTML/CSS: http://mylinux.langara.bc.ca/~walvarez00/1045/SnapPaint%20Project/index.html

HTML Validation: https://validator.w3.org/nu/?doc=http%3A%2F%2Fmylinux.langara.bc.ca%2F~walvarez00%2F1045%2FSnapPaint%2520Project%2Findex.html

**View / Controller Design**

**Primary User Controls:**

//Filter Buttons event handler for all 4 filter buttons

filterOneButton.onclick = applyFilter();

fIlterTwoButton.onclick = applyFilter();

filterThreeButton.onclick = applyFilter();

filterFourButton.onclick = applyFilter();

//Meme Buttons event handler for all 4 meme buttons

memeButtonOne.onclick = applyMeme():

memeButtonTwo.onclick = applyMeme();

memeButtonThree.onclick = applyMeme();

memeButtonFour.onclick = applyMeme();

//Load Image from computer event handlers

computerImage.onclick = loadImageButton();

computerImage.onload = loadImage();

//Input text and have it display on top of image event handlers

textInput.oninput = inputText():

//Save Image

saveImage.onclick = saveImage();

**SnapPaint View:**

//Apply filter to image using css filter effects

function applyFIlter(filter);

//Overlay image on top of loaded image

function applyMeme(image);

//Load image from computer

function loadImage();

//Button click that triggers the loadImage(); function

function loadImageButton();

//Have user input text and display on top of image

function inputText();

//Save the modified image

function saveImage();

**View / Controller Design**

Key Objects: Filters, Meme Images, and Text Input

Filter {

setFilter(filter); // function that sets a filter property to the filter object

}

Meme Images {

setImage(image); // function that sets an image property to the image object

}